

# Jira and Slack Integration with Jenkins

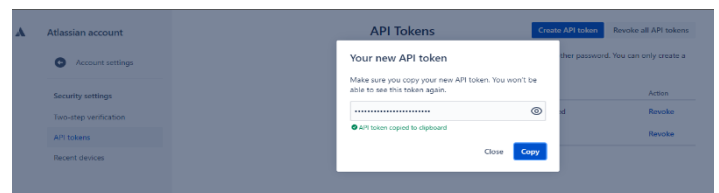
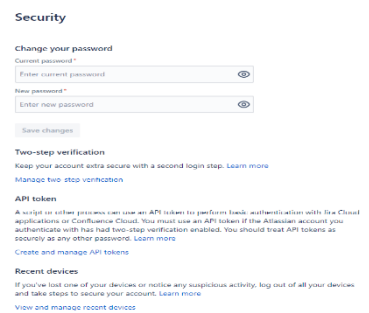
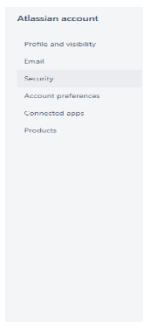
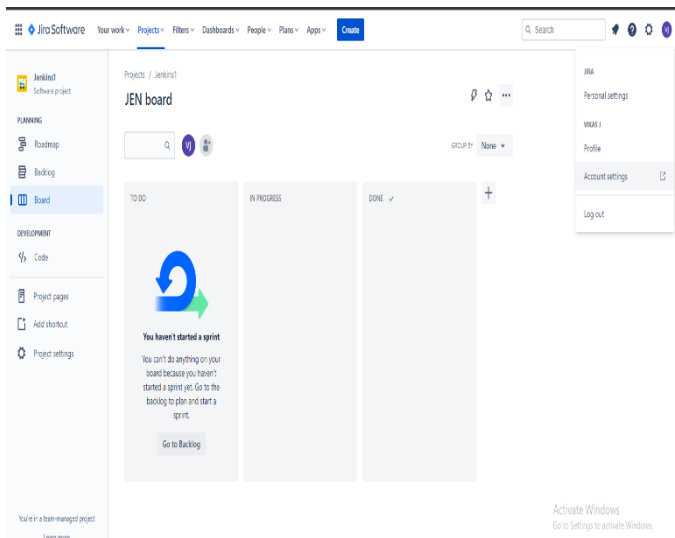
## 1) Installation and Configuration of Jenkins

- 1) `sudo yum install java-11`  
or  
for redhat "`sudo yum install java-11-openjdk-devel`"
  - 2) Install and config Maven
    - `cd /opt`
    - `sudo wget https://dlcdn.apache.org/maven/maven-3/3.8.5/binaries/apache-maven-3.8.5-bin.zip`
    - `sudo unzip apache-maven-3.8.5-bin.zip`
    - `cd ~`
    - `ll -a`
    - `sudo vi .bashrc`
    - Paste the following "`export PATH=/opt/apache-maven-3.8.5/bin:$PATH`" and save and exit
    - `source .bashrc`
    - `mvn -v`
  - 3) `sudo yum install git`
  - 4) Download jenkins package
    - for amazon linux "`sudo amazon-linux-extras install epel`"
    - or
    - for redhat "`sudo dnf install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm`"
    - `sudo wget -O /etc/yum.repos.d/jenkins.repo https://pkg.jenkins.io/redhat-stable/jenkins.repo`
    - `sudo rpm --import https://pkg.jenkins.io/redhat-stable/jenkins.io.key`
  - 5) Install Jenkin
    - `sudo yum install jenkins`
  - 6) Start Jenkins
    - `sudo systemctl start jenkins`
  - 7) check status of jenkins if required
    - `systemctl status jenkins.service`
- Note:** By default port no 8080 will be assigned for jenkins.
- 8) Change the port if required
    - `sudo vi /etc/sysconfig/jenkins` (port 8090)
    - `sudo systemctl restart Jenkins`

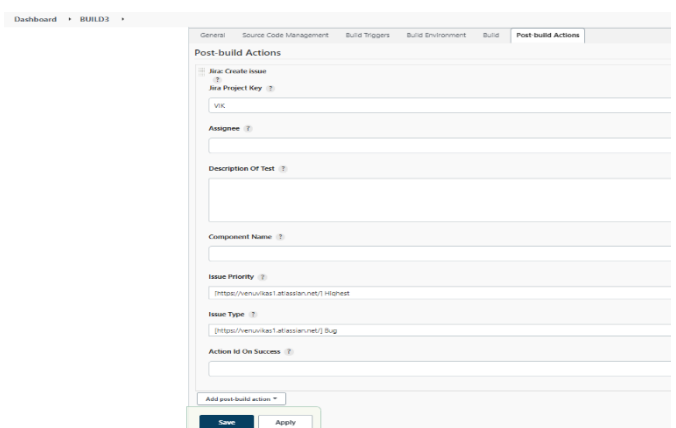
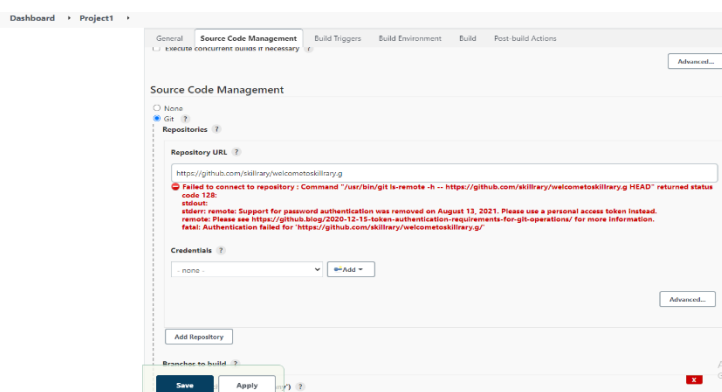
## 2) Plugins required to integrate with Jenkins

- **Slack Notification**
- **Jira**

- 1) SignUp/SignIn to Jira, Create Project and note down Project key. In this eg. it is “JEN”.
- 2) Click on Account Settings → Security → Create and Manage API Token → Create a new API Token and save the keyphrase generated.



- 3) Install Jira Plugin in Jenkins.
- 4) Create Free style Project in Jenkins
- 5) Here git url is pasted wrong to make build fail.
- 6) Goto Post build action → Select Jira:Create issue → Select issue priority → Select issue type → Save & Build.



**Result :** If the build fails then it will rise a ticket in Jira.

### 3) Integrating Slack with Jenkins

- SignUp/SignIn to slack and create a workspace as well as channel.
- Go to <https://api.slack.com> → Create a Slack App, choose workspace → Save.
- Select OAuth & Permission → Scopes → Add an OAuthScope → Select chat:write → Scroll Up: Inside App to workspace → Allow → Copy the token generated.
- Goto Channel → in chat box “@appname” → Invite to channel.
- Goto Jenkins → In manage credentials add the token generated from slack by selecting as “Secret Text”.
- Install “Slack Notification Plugin” in Jenkins.
- Inside configure system → go to Slack settings → select “Custom slack app bot user” → Save.
- Project configuration → Post build actions → Select “Slack Notification” → Provide workspace name, credentials, channel name and then test connection.

**Result** : Build status of Project will be automatically listed in slack as a chat

The image displays two screenshots related to integrating Slack with Jenkins. The top screenshot shows the Slack API 'OAuth & Permissions' page for an app named 'App1'. The left sidebar lists various settings and features, with 'OAuth & Permissions' selected. The main content area includes a warning about token rotation, an 'Opt In' button, and instructions for installing the app to a workspace. The bottom screenshot shows the Jenkins 'Configure' page for a build named 'BUILD3'. The 'Post-build Actions' tab is active, and a list of actions is shown, with 'Slack Notifications' highlighted. A configuration modal for 'Slack Notifications' is open, showing fields for 'Workspace' (set to 'SkillRary'), 'Credential' (set to 'Slack'), 'Channel / member id' (set to '#proj1'), and 'Success'.